PATENT ABSTRACTS OF JAPAN

(11)Publication number : **09-284767**

(43)Date of publication of application : **31.10.1997**

(51)Int.Cl. H04N 7/30
H03M 7/30
H04N 1/41

(21)Application number : **08-246555**

(22)Date of filing : **18.09.1996**

(71)Applicant : **FUJI FILM MICRO DEVICE KK**
FUJI PHOTO FILM CO LTD

(72)Inventor : **ASANO MASANARI**

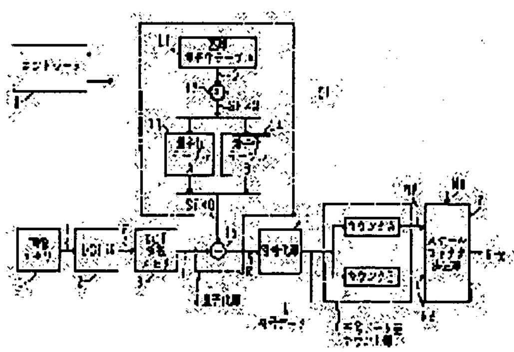
(30)Priority

Priority number : **08 25202** Priority date : **13.02.1996** Priority country : **JP**

(54) PICTURE COMPRESSION SYSTEM AND PICTURE COMPRESSION METHOD

(57)Abstract:

PROBLEM TO BE SOLVED: To accelerate a compression processing by parallelly performing the process of preparing code data by the two kinds of compression degrees after generating the DCT coefficient of a block 1 and the process of generating the DCT coefficient of the block 2.



SOLUTION: A DCT part 2 performs a DCT processing to the picture data of the block 1 first, and when a controller 8 detects the end, a similar processing is performed to the blocks 2-(n). Then, for the DCT coefficient generated by ending the DCT processing to the block 1, the controller 8 instructs a quantization part 4 to perform quantization in the quantization table 11 of a scale factor SFx and quantized data are generated. When the controller 8 detects generation end, an encoding processing is instructed to an encoding part 5 and the code data of the block 1 are generated. Similarly thereafter, the processing is continued until the last block (n) and the picture data of one frame are compressed.

LEGAL STATUS

[Date of request for examination]	06.10.2000
[Date of sending the examiner's decision of rejection]	05.08.2003
[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]	
[Date of final disposal for application]	
[Patent number]	
[Date of registration]	
[Number of appeal against examiner's decision of rejection]	
[Date of requesting appeal against examiner's decision of rejection]	
[Date of extinction of right]	

CLAIMS

[Claim(s)]
[Claim 1] An image data supply means to divide per block and to supply image data (1), A DCT means to carry out the discrete cosine transform (DCT) of the image data of the block unit supplied from said image data supply means, and to generate the DCT multiplier of a block unit (2), A code data generation means to generate code data continuously with the 1st and 2nd condensation about the DCT multiplier whenever said DCT means generates the DCT multiplier of one block (4 5), The counter which accumulates the amount of the code data about each of the 1st and 2nd condensation which said code data generation means generates about all blocks supplied (6), The picture compression system which has a condensation presumption means (7) to presume the condensation for generating the code data of the target amount of data according to the amount of code data of both of the 1st and 2nd condensation generated by said counter.
[Claim 2] Furthermore, the picture compression system according to claim 1 which has a means to direct to generate code data with the condensation which said condensation presumption means presumes for said code data generation means.
[Claim 3] The code data of the 1st and 2nd condensation is a picture compression system according to claim 1 or 2 generated by quantizing using the 1st and 2nd quantization tables, respectively including a quantization means for said code data generation means to quantize a DCT multiplier.
[Claim 4] Said code data generation means is a picture compression system including a means to generate the 1st and 2nd quantization tables according to claim 3, by multiplying a basis child-ized table by the 1st and 2nd scale factors.
[Claim 5] Said code data generation means is a picture compression system including a coding means to perform Huffman coding after quantizing a DCT multiplier according to claim 3 or 4.
[Claim 6] It is the picture compression system according to claim 1 to 5 which completes generation of the code data of both of the 1st and 2nd condensation about the block before one before said DCT means is a means which carries out DCT of the image data continuously per block and said code data generation means completes DCT of the image data about the block with said DCT means.
[Claim 7] Said image data supply means is a picture compression system according to claim 1 to 6 which supplies the image data about the sample block of the images of one frame.
[Claim 8] The process which carries out the discrete cosine transform (DCT) of the image data of the 1st block, and generates the DCT multiplier of the 1st block, The process which carries out DCT of the image data of the 2nd block, and generates the DCT multiplier of the 2nd block after generation of the DCT multiplier of said 1st block is completed, After generation of the DCT multiplier of said 1st block is completed, by the time generation of the DCT multiplier of said 2nd block is completed The picture compression approach including the process which generates code data, continues with the 1st condensation about the DCT multiplier of this 1st block, and generates code data with the 2nd condensation.

[Translation done.]

DETAILED DESCRIPTION

[Detailed Description of the Invention]
[0001]
[Field of the Invention] Especially this invention relates to the picture compression system and the picture compression approach of compressing a digital image and lessening the amount of data about a digital image processing system.
[0002]
[Description of the Prior Art] Although a picture compression system is used, a digital still camera is in one. A digital still camera photos a digital static image by turning a lens to a photographic subject and pushing a shutter carbon button. The data compression of the image by which image formation is carried out through a lens is changed and carried out to an electrical signal, and it is memorized by the memory card which can be renewed. A data compression is processing for reducing the amount of data and making a memory card memorize many image data.
[0003] The amount of the code data obtained by carrying out the data compression of the digital image changes with spatial frequency distribution which a digital image has. For example, the amount of code data can seldom be lessened about the digital image containing many high frequency components. On the other hand, about a digital image with few high frequency components, the amount of code data can be lessened considerably. That is, although it changes with methods of a data compression, the amount of the code data generally generated by the data compression changes with classes of digital image.
[0004] The code data by which the data compression was carried out is memorized by storages, such as a memory card. A memory card has the memory capacity of 1 M byte, and cannot make 1 M bytes or more of data memorize in that case.
[0005] In order to make it not write code data in a memory card more than 1 M byte, it is necessary to tell a photography person about the remaining number of sheets recordable for a photography person's facilities. The code data by which a data compression is carried out is not based on the class of digital image, but if it is all the same amount of data per each image, a photography person can be easily told about the number of sheets of a digital image recordable on a memory card.
[0006] However, when the amount of code data is adjustable, a photography person cannot be told about the remaining number of sheets. If there are few amounts of code data of the image to be photoed from now on, many number of sheets is recordable, and if there are many amounts of code data of the image to photo, only small number of sheets is recordable.
[0007] So, in case the data compression of the digital image is carried out, it is necessary to perform fixed-length-ized processing of code data. By performing fixed-length-ized processing, even if it is what kind of digital image, it is convertible for the code data of about 1 quantum. Fixed-length-ized processing is processing for carrying out the data compression of the digital image of one sheet (one frame), and generating fixed-length code data. If code data is a fixed length, a photography person can be told about the remaining number of sheets.
[0008] Next, fixed-length-ized processing is explained. In order to perform fixed-length-ized processing, statistics processing is first performed as pretreatment, the condensation of a data compression is

adjusted according to the result of the statistics processing, and fixed-length code data is generated.

[0009] A photography person's push of a shutter carbon button incorporates a digital image. Next, statistics processing is performed to the incorporated digital image. Statistics processing is processing which guesses the code data of the amount of which about is generated, when it compresses about the incorporated digital image.

[0010] Termination of statistics processing performs compression processing and storage processing. as a result of statistics processing, more code data are generated and meet -- what is necessary is just to compress by setting up condensation more highly, if it is surmised that it comes out code data is generated fewer and meets -- what is necessary is just to compress by setting up condensation lowness, if it is surmised that it comes out The code data generated by the data compression serves as the always almost fixed amount of data.

[0011] Then, the code data by which the data compression was carried out is recorded on a memory card by storage processing. Above, a series of processings to the record to a memory card from incorporation of a digital image are ended.

[0012]

[Problem(s) to be Solved by the Invention] In picture compression processing, there are the following approaches as an approach of performing fixed-length-ized processing. First, compression processing is performed once with the condensation of criteria as statistics processing. Consequently, when more amounts of data than the target amount of data are generated, it is set as condensation higher than the condensation of criteria. On the other hand, when the code data of an amount smaller than the target amount of data is generated, it is set as condensation lower than the condensation of criteria. With the set-up condensation, formal compression processing is performed and the code data of an image is generated.

[0013] However, since the relation between condensation and the amount of code data is not fixed, dispersion produces it according to the class of image for the error of the target amount of data and the amount of code data generated. Only by performing statistics processing once, the precision of fixed-length-izing is low.

[0014] In order to perform highly precise fixed-length-ization, statistics processing is performed twice or more and there is a method of deciding condensation. The case where 2 times is performed for statistics processing is explained to an example. First, as the 1st statistics processing, compression processing is performed with the 1st criteria condensation, and the amount of code data is estimated. Next, compression processing is performed with the 2nd different criteria condensation from the 1st criteria condensation as the 2nd statistics processing, and the amount of code data is estimated. Formal condensation is set up in consideration of the above two amounts of code data. For example, what is necessary is just to set up formal condensation between the 1st criteria condensation and the 2nd criteria condensation, when the target amount of data is between the two amounts of code data concerned. With the set-up condensation, formal compression processing is performed and the code data of an image is generated.

[0015] However, by this approach, in order to perform statistics processing twice and to perform formal compression processing once, a total of three compression processings will be performed and great time amount is needed.

[0016] The purpose of this invention is offering the picture compression system which performs a high speed and highly precise fixed-length-ized processing.

[0017] Other purposes of this invention are offering the picture compression approach of performing a high speed and highly precise fixed-length-ized processing.

[0018]

[Means for Solving the Problem] An image data supply means to divide the picture compression system of this invention per block, and to supply image data, A DCT means to carry out the discrete cosine transform (DCT) of the image data of the block unit supplied from said image data supply means, and to generate the DCT multiplier of a block unit, A code data generation means to generate code data continuously with the 1st and 2nd condensation about the DCT multiplier whenever said DCT means

generates the DCT multiplier of one block, The counter which accumulates the amount of the code data about each of the 1st and 2nd condensation which said code data generation means generates about all blocks supplied, According to the amount of code data of both of the 1st and 2nd condensation generated by said counter, it has a condensation presumption means to presume the condensation for generating the code data of the target amount of data.

[0019] The process which the picture compression approach of this invention carries out the discrete cosine transform (DCT) of the image data of the 1st block, and generates the DCT multiplier of the 1st block, The process which carries out DCT of the image data of the 2nd block, and generates the DCT multiplier of the 2nd block after generation of the DCT multiplier of said 1st block is completed, The process which will generate code data, will continue with the 1st condensation about the DCT multiplier of this 1st block, and will generate code data with the 2nd condensation by the time generation of the DCT multiplier of said 2nd block is completed, after generation of the DCT multiplier of said 1st block is completed is included.

[0020] If a DCT means finishes generating the DCT multiplier of the 1st block, about the DCT multiplier concerned, a code data generation means will generate code data, will continue with the 1st condensation, and will generate code data with the 2nd condensation. A DCT means can generate the DCT multiplier of the 2nd block, after generating the DCT multiplier of the 1st block. If a DCT means finishes generating the DCT multiplier of the 1st block, the process at which a code data generation means generates code data with at least two kinds of condensation, and the process at which a DCT means generates the DCT multiplier of the 2nd block can be made to parallelize. By making the process concerned parallelize, improvement in the speed of compression processing can be attained. Moreover, highly precise fixed-length-ized processing can be performed by generating code data with at least two kinds of condensation.

[0021]

[Embodiment of the Invention] Drawing 2 is a flow chart which shows the procedure which the picture compression system by the example of this invention performs.

[0022] Statistics processing is performed at step S1. This statistics processing computes the amount of code data about two kinds of condensation between 1 time of pressing time. Usually, in order to compute the amount of code data about two kinds of condensation, the time amount of two compression processings of the minimum is needed. The description of statistics processing here is generating efficiently the code data about two kinds of condensation, and processing for a short time. After generating code data, formal condensation is determined based on the amount of two kinds of code data concerned.

[0023] Step S2 performs formal compression processing with the condensation determined by statistics processing. The code data of an image is generated by this compression processing. The amount of the code data generated can be brought close to the target amount of data with high degree of accuracy. Above, fixed-length compression processing is ended.

[0024] Drawing 1 is the block diagram showing the picture compression structure of a system by this example. This picture compression system generates the code data of the JPEG (joint photographic expert group) method which is a standard compression method of a digital static image. The resource of the system of the conventional JPEG method is utilizable as it is.

[0025] A picture compression system has an image memory 1, the discrete cosine transform (henceforth DCT) section 2, the DCT coefficient memory 3, the quantization section 4, the coding section 5, the amount count area 6 of code data, the scale-factor decision section 7, and a controller 8. A controller 8 delivers a timing signal among other the processing blocks of all, and adjusts the timing during a processing block.

[0026] Next, each processing block is explained. Image memories 1 are DRAM and a flash memory, and memorize the image data of one frame. In the image memory 1, image data is usually memorized in the raster format. Image data consists of two or more pixel data.

[0027] Raster formats are the following pixel data lists about the image of one frame. First, it begins from the pixel of the upper left corner of an image, and stands in a line one by one toward a right

horizontal direction. If it carries out to a right end pixel, it will begin from the pixel at the left end of next Rhine, and will stand in a line one by one toward a right horizontal direction. Hereafter, it carries out to Rhine under No. 1 similarly. The pixel of a lower right corner serves as the last data.
[0028] In order that a picture compression system may process for every 8x8-pixel block fundamentally, an image memory 1 changes image data into a block type from a raster format, and supplies it to the DCT section 2. The number of image data of monochrome image is one. Although a color picture separates to brightness data and color data, it performs a raster / block conversion for each as another image data.
[0029] Block types are the following pixel data lists about the image of one frame. Field division of the image of one frame is carried out at two or more blocks. 1 block is 8x8 pixels. Like the above-mentioned raster format, the sequence of the block in one frame begins from the block of an upper left corner, and is located in a line with a right horizontal direction. The last block is a block of a lower right corner. The pixel data list within a block is the same as that of a raster format too, begins from the pixel data of the upper left corner within a block, and is located in a line with a right horizontal direction. The last pixel data are pixel data of the lower right corner within a block.
[0030] Image data I of a block type is supplied to the DCT section 2. Hereafter, processing is performed by making 1-block image data into one unit. That is, JPEG compression divides the image of one sheet into a 8x8-pixel block, and carries out the following processings for the block concerned to a unit.
[0031] The DCT section 2 performs DCT processing about image data [of a block unit] I. DCT processing is the transposition cosine coefficient matrix Dt about image data I. The DCT multiplier F is obtained by inserting by the cosine coefficient matrix D and performing matrix operation.
[0032] F=Dt ID -- here, the DCT multiplier F is the matrix of 8x8, and shows a spatial-frequency component.
[0033] The DCT coefficient memories 3 are DRAM and SRAM, and memorize the DCT multiplier F generated in the DCT section 2.
[0034] Next, the configuration of the quantization section 4 is explained. Memory 11 memorizes the basis child-ized table Q.
[0035] Drawing 3 shows the example of the basis child-ized table Q. As mentioned above, since a picture compression system performs a data compression per block of 8x8, the quantization table Q is constituted by the matrix of 8x8 corresponding to it.
[0036] The basis child-ized table Q is a quantization table for performing a data compression with standard condensation. Quantization processing does a division to the DCT multiplier F of 8x8 by the multiplier to which it corresponds in the quantization table Q. A DCT multiplier has a low spatial frequency component, and its frequency component is as high as the direction of the upper left of a matrix as the direction of the lower right. The basis child-ized table Q is as fine as a frequency component low as a whole, and it is shown that a higher frequency component quantizes coarsely. Generally, a data compression is performed by deleting the information on the high frequency component of image data in consideration of a high frequency component having many noises in consideration of human being's vision property.
[0037] In drawing 1 , a multiplier 12 multiplies the basis child-ized table Q by the scale factor SF. That is, all the elements of the matrix of the basis child-ized table Q are multiplied by the scale factor SF. A multiplier 12 outputs quantization table SF-Q. Quantization table SF-Q is memorized by either memory 13 or the memory 14.
[0038] The quantization table A is memorized by memory 13 and the quantization table B is memorized by memory 14. Scale factors SF differ on the quantization table A and the quantization table B. The condensation of code data is decided by the value of a scale factor SF. That is, the difference between the quantization table A and the quantization table B shows the difference in condensation. The quantization table A is expressed with Q-SFa, and the quantization table B is expressed with Q-SFb.
[0039] Either of the quantization quantization tables A or B is supplied to a divider 15 as SF-Q. In case statistics processing is performed, the quantization table A is used as 1st criteria table, and the quantization table B is used as 2nd criteria table.

[0040] A divider 15 breaks the DCT multiplier Fuv memorized by the DCT coefficient memory 3 by quantization table SF-Quv, and outputs the quantization multiplier Ruv. Rolling round means integer-ization to the nearest integer.
[0041]
Ruv=round [Fuv/(SF-Quv)]
The coding section 5 performs coding processing to the quantization data Ruv. Coding processing includes processing of run length coding and Huffman coding. Run length coding can perform high compression to data with which the value of 0 continues continuously. For the quantization data Ruv, many 0 is an assembly and a cone to the lower right part (high frequency component) of a matrix. High compression can be performed if run length coding is performed for the matrix Ruv of quantization data with a zigzag scan using this property. A zigzag scan is the approach of performing a sequential scan towards a high frequency component from a low-frequency component.
[0042] After the coding section 5 performs run length coding, it performs Huffman coding and generates code data.
[0043] The amount count area 6 of code data has Counter A and Counter B. Counter A counts the amount NA of code data when quantizing on the quantization table A and generating code data. Counter B counts the amount NB of code data when quantizing on the quantization table B and generating code data.
[0044] In other words, Counter A counts the amount NA of code data when compressing with the 1st criteria condensation (quantization table A) at the time of statistics processing. Counter B counts the amount NB of code data when compressing with the 2nd criteria condensation (quantization table B).
[0045] Drawing 4 shows how the amount count area 6 of code data counts the amount of code data. The image of one frame consists of n blocks. Code data is generated per block. Counter A and Counter B accumulate the amount of the code data of all blocks (n blocks), and compute the amount of the code data of the image of one frame.
[0046] Counter A computes the amount NA of code data when compressing using the quantization table A. Counter B computes the amount NB of code data when compressing using the quantization table B.
[0047] Here, the scale factor SFa about the quantization table A shows an example when larger than the scale factor SFb about the quantization table B. That is, if the quantization table A is used, high compression can be performed rather than the time of using the quantization table B.
[0048] The amount NA of code data when compressing about the same image data using the quantization table A becomes smaller than the amount NB of code data when compressing using the quantization table B.
[0049] In drawing 1 , the scale-factor decision section 7 determines the formal scale factor SFx based on two kinds of amounts NA and NB of code data. A scale factor SFx is presumed as condensation for generating the code data of the target amount of data NX. If a scale factor SFx can be found, statistics processing (step S1 of drawing 2) will be ended.
[0050] After statistics processing is completed, compression processing (step S2 of drawing 2) is performed. In compression processing, a scale factor SFx is inputted into the quantization section 4 as a scale factor SF. If it compresses using a scale factor SFx, the code data near the target amount of data NX is generable.
[0051] Drawing 5 is a graph for explaining processing of the scale-factor decision section 7. An axis of ordinate is a scale factor and an axis of abscissa is the amount of code data. The scale-factor decision section 7 holds a scale factor as a function of the amount of code data. As an example, the case where a scale factor is proportional to the amount of code data is explained.
[0052] When it compresses using a scale factor SFa, the amount NA of code data is obtained. When it compresses using a scale factor SFb, the amount NB of code data is obtained. Based on these data, a property (it approximates in a straight line) as shown in drawing 5 is set up.
[0053] The target amount of data NX is directed from the outside. The scale factor corresponding to this target amount of data is searched for from an above-mentioned property. In order to obtain the target amount of data NX, it turns out that what is necessary is just to compress using a scale factor SFx.

[0054] From drawing 5 , a scale factor SFx is the following, and can be made and searched for.
(NB-NA):(NX-NA)=(SFa-SFb):(SFa-SFx)
(NB-NA)x(SFa-SFx)=(NX-NA)x(SFa-SFb)
(SFa-SFx)=(NX-NA)x(SFa-SFb)/(NB-NA)
SFx=SFa-{(NX-NA)x(SFa-SFb)/(NB-NA)}
[0055] In addition, a scale factor SFx is searched for using an upper type, and also a scale factor SFx may be searched for by the approximate expression of an upper type. Moreover, it asks by the formula and also a scale factor SFx may be determined using a look-up table. Furthermore, not only proportionality but an inverse proportion etc. can also set the relation between a scale factor and the amount of code data as other relation. However, it is desirable that a scale factor SFx can be determined with a short time and an easy configuration.
[0056] Moreover, in case compression processing is performed using a scale factor SFx and code data is generated, in order to make it the amount of code data concerned not exceed the target amount of data, you may make it multiply further the scale factor searched for above by the safety factor.
[0057] Next, in the picture compression system of drawing 1 , it divides into statistics processing and compression processing, and each procedure is explained.
[0058] First, statistics processing is explained.
(1) Statistics processing statistics processing computes two kinds of amounts NA and NB of code data by the time amount of one compression processing. This is closed if the processing time in the DCT section 2 is overwhelmingly possible in compression processing using the property of a ********.
[0059] The processing time in the DCT section 2 forms 5 - 80 percent of the whole. The time amount taken to carry out quantization processing (quantization section 4) to the time amount which takes 1 block (8x8 pixels) image data to carry out DCT processing (DCT section 2) is compared.
[0060] The following counts are calculated in the DCT section 2. Here, the case where the usual DCT algorithm is used is shown.
[0061] Multiplication 1024 times addition In the quantization section 4, the following counts are calculated 896 times.
[0062] Multiplication 64 (= 8x8) time, as mentioned above, there are many counts of an operation which the DCT section 2 performs extraordinarily compared with the quantization section 4, and its processing time is overwhelmingly long.
[0063] Drawing 6 is a timing chart which shows the timing to which the picture compression system of drawing 1 performs statistics processing. An axis of abscissa shows time amount.
[0064] The DCT section 2 processes in order of processing alpha 1, processing alpha 2, and processing alpha 3. Processing alpha 1 is DCT processing performed about the image data of block 1 (1st block). It is the DCT processing which performs processing alpha 2 about the image data of block 2 (2nd block), and performs processing alpha 3 about the image data of block 3 (3rd block).
[0065] The DCT section 2 starts the DCT processing alpha 1 about the image data of block 1 in time of day t0. In time of day t10, a controller 8 directs initiation of the DCT processing alpha 2 about the image data of block 2 in the DCT section 2, if termination of the DCT processing alpha 1 is detected. Then, in time of day t20, detection of termination of the DCT processing alpha 2 directs initiation of the DCT processing alpha 3 about the image data of block 3 in the DCT section 2. The DCT processing alpha 3 is completed in time of day t30. The DCT section 2 processes image data continuously from block 1 to Block n (the last block).
[0066] Next, processing of the quantization section 4 and the coding section 5 is explained. In time of day t10, a controller 8 directs to start quantization processing beta1A about the DCT multiplier F generated by the DCT processing alpha 1 in the quantization section 4, if termination of the DCT processing alpha 1 about image data [of block 1] I is detected. The quantization section 4 performs quantization processing beta1A using the quantization table A. Quantization processing beta1A is processing which quantizes the DCT multiplier F of block 1 on the quantization table A.
[0067] In time of day t11, a controller 8 directs to start coding processing gamma1A about the quantization data R generated by quantization processing beta1A in the coding section 5, if termination

of quantization processing beta1A is detected. Coding processing gamma1A is coding processing about data in which the quantization table A was used about the block 1.

[0068] In time of day t12, termination of coding processing gamma1A generates the code data based on having used the quantization table A. Then, although not illustrated to drawing 6 , the counter A of the amount count area 6 (drawing 1) of code data counts the amount of the code data.

[0069] The above performed quantization processing and coding processing about the quantization table A. Next, quantization processing and coding processing are performed about the quantization table B.

[0070] In time of day t12, a controller 8 directs to start quantization processing beta1B about the DCT multiplier F generated by the DCT processing alpha 1 in the quantization section 4, if termination of coding processing gamma1A is detected. However, a controller 8 directs to perform quantization processing beta1B using the quantization table B in that case. Quantization processing beta1B is processing which quantizes the DCT multiplier F of block 1 on the quantization table B.

[0071] In time of day t13, a controller 8 directs to start coding processing gamma1B about the quantization data R generated by quantization processing beta1B in the coding section 5, if termination of quantization processing beta1B is detected. Coding processing gamma1B is coding processing about data in which the quantization table B was used about the block 1.

[0072] In time of day t14, termination of coding processing gamma1B generates the code data based on having used the quantization table B. Then, although not illustrated to drawing 6 , the counter B of the amount count area 6 (drawing 1) of code data counts the amount of the code data.

[0073] In time of day t12, generation of the code data based on the quantization table A is completed, and generation of the code data based on the quantization table B is completed in time of day t14. Then, the DCT processing alpha 2 is completed in time of day t20. That is, while performing DCT processing alpha 2, quantization processing beta1A, beta1B, and coding processing gamma1A and gamma1B are completed.

[0074] The above performed quantization processing and coding processing about the block 1. Next, quantization processing and coding processing are performed about block 2.

[0075] In time of day t20, a controller 8 directs to start quantization processing beta2A about the DCT multiplier F generated by the DCT processing alpha 2 in the quantization section 4, if termination of the DCT processing alpha 2 about image data [of block 2] I is detected. A controller 8 directs to perform quantization processing beta2A using the quantization table A in that case. Quantization processing beta2A is processing which quantizes the DCT multiplier F of block 2 on the quantization table A.

[0076] In time of day t21, a controller 8 directs to start coding processing gamma2A about the quantization data R generated by quantization processing beta2A in the coding section 5, if termination of quantization processing beta2A is detected. Coding processing gamma2A is coding processing about data in which the quantization table A was used about the block 2.

[0077] In time of day t22, termination of coding processing gamma2A generates the code data based on having used the quantization table A. Then, although not illustrated to drawing 6 , the counter A of the amount count area 6 (drawing 1) of code data counts the amount of the code data.

[0078] Next, quantization processing and coding processing are performed about the quantization table B. In time of day t22, a controller 8 directs to start quantization processing beta2B about the DCT multiplier F generated by the DCT processing alpha 2 in the quantization section 4, if termination of coding processing gamma2A is detected. However, a controller 8 directs to perform quantization processing beta2B using the quantization table B in that case. Quantization processing beta2B is processing which quantizes the DCT multiplier F of block 2 on the quantization table B.

[0079] In time of day t23, a controller 8 directs to start coding processing gamma2B about the quantization data R generated by quantization processing beta2B in the coding section 5, if termination of quantization processing beta2B is detected. Coding processing gamma2B is coding processing about data in which the quantization table B was used about the block 2.

[0080] In time of day t24, termination of coding processing gamma2B generates the code data based on having used the quantization table B. Then, although not illustrated to drawing 6 , the counter B of the amount count area 6 (drawing 1) of code data counts the amount of the code data.

[0081] In time of day t22, generation of the code data based on the quantization table A is completed, and generation of the code data based on the quantization table B is completed in time of day t24. Then, the DCT processing alpha 3 is completed in time of day t30. While performing DCT processing alpha 3, quantization processing beta2A, beta2B, and coding processing gamma2A and gamma2B are completed.

[0082] Hereafter, processing is similarly continued to the last block n. The time amount of statistics processing of this single string is equivalent to the time amount in which the DCT section 2 carries out DCT processing about all blocks, i.e., the time amount of compression processing of one usual batch.
[0083] Though the time amount of statistics processing is the time amount of compression processing of one batch, a picture compression system can generate two kinds of code data about the quantization table A and the quantization table B, and can compute the amounts NA and NB of code data.
[0084] Drawing 6 showed the timing in each processing section. Next, the timing of the processing for every block is shown.
[0085] Drawing 7 is a timing chart which shows the timing of a block unit which performs statistics processing. An axis of abscissa shows time amount and supports the time amount of drawing 6 .
[0086] First, processing of block 1 is explained. In time of day t0, the DCT processing alpha 1 about the image data of block 1 begins, and the DCT processing alpha 1 is completed in time of day t10. After the DCT processing alpha 1 is completed, generation processing 51A of the code data using the quantization table A is performed, and generation processing 51B of the code data using the quantization table B is performed after that.
[0087] Processing 51A contains quantization processing beta1A which used the quantization table A, and coding processing gamma1A of the quantization data generated by the quantization processing beta1A concerned. Processing 51B contains quantization processing beta1B which used the quantization table B, and coding processing gamma1B of the quantization data generated by the quantization processing beta1B concerned.
[0088] Next, processing of block 2 is explained. In time of day t10, the DCT processing alpha 2 about the image data of block 2 begins, and the DCT processing alpha 2 is completed in time of day t20. After the DCT processing alpha 2 is completed, generation processing 52A of the code data using the quantization table A is performed, and generation processing 52B of the code data using the quantization table B is performed after that. Processing 51A contains quantization processing beta2A and coding processing gamma2A. Processing 51B contains quantization processing beta2B and coding processing gamma2B.
[0089] As mentioned above, since the processing time of DCT is long, two quantization processings and coding processing (for example, 51A and 51B) can be performed between the DCT processings (for example, alpha 2) about 1 block. Thereby, two statistics processings can be substantially performed among one compression processing about the image of one frame. As for one statistics processing, the amount NA of code data is computed using the quantization table A. As for another statistics processing, the amount NB of code data is computed using the quantization table B.
[0090] Statistics processing searches for the formal scale factor SFx, after computing the amounts NA and NB of code data. Termination of statistics processing performs compression processing using a scale factor SFx. Next, compression processing is explained.
(2) Compression processing drawing 8 is a timing chart which shows the timing to which the picture compression system of drawing 1 </A> performs compression processing . An axis of abscissa shows time amount.
[0091] The DCT section 2 starts the DCT processing alpha 1 about the image data of block 1 in time of day t50 first. In time of day t60, a controller 8 directs initiation of the DCT processing alpha 2 about the image data of block 2 in the DCT section 2, if termination of the DCT processing alpha 1 is detected. Then, in time of day t70, detection of termination of the DCT processing alpha 2 directs initiation of the DCT processing alpha 3 about the image data of block 3 in the DCT section 2. The DCT processing alpha 3 is completed in time of day t80. The DCT section 2 processes image data continuously from block 1 to Block n.

[0092] Next, processing of the quantization section 4 and the coding section 5 is explained. In time of day t60, a controller 8 directs to start the quantization processing beta 1 about the DCT multiplier generated by the DCT processing alpha 1 in the quantization section 4, if termination of the DCT processing alpha 1 about the image data of block 1 is detected. A controller 8 directs to perform quantization processing beta 1 using the quantization table of a scale factor SFx in that case.
[0093] In time of day t61, a controller 8 directs to start the coding processing gamma 1 about the quantization data generated by the quantization processing beta 1 in the coding section 5, if termination of the quantization processing beta 1 is detected. The coding section 5 starts generation of the code data about block 1. The coding processing gamma 1 is completed at time of day t62.
[0094] The above performed quantization processing and coding processing about the block 1. Next, quantization processing and coding processing are performed about block 2.
[0095] In time of day t70, a controller 8 directs to start the quantization processing beta 2 about the DCT multiplier generated by the DCT processing alpha 2 in the quantization section 4, if termination of the DCT processing alpha 2 about the image data of block 2 is detected. A controller 8 directs to perform quantization processing beta 2 using the quantization table of a scale factor SFx in that case.
[0096] In time of day t71, a controller 8 directs to start the coding processing gamma 2 about the quantization data generated by the quantization processing beta 2 in the coding section 5, if termination of the quantization processing beta 2 is detected. The coding section 5 starts generation of the code data about block 2. The coding processing gamma 2 is completed at time of day t72.
[0097] Hereafter, similarly, processing is continued to the last block n and the image data of one frame is compressed. Above, statistics processing and compression processing are ended.
[0098] Drawing 9 shows other examples of the amount count area 6 of code data of drawing 1 . The amount count area 6 of code data counts the amount NA of code data about the quantization table A, and the amount NB of code data about the quantization table B.
[0099] The code data count area 6 has a counter 61, Register A, and Register B. A counter 61 counts the amount of the code data about the quantization table A, and the amount of the code data about the quantization table B by time sharing under control of the controller 8 of drawing 1 .
[0100] The amount of code data of the block unit about the quantization table A is added to Register A. When the addition about all blocks is completed, the amount NA of code data is memorized by Register A. The amount NA of code data is the amount of code data of the image of one frame when using the quantization table A.
[0101] On the other hand, the amount of code data of the block unit about the quantization table B is added to Register B. When the addition about all blocks is completed, the amount NB of code data is memorized by Register B. The amount NB of code data is the amount of code data of the image of one frame when using the quantization table B.
[0102] Drawing 11 shows other examples of the quantization table 4 of drawing 1 . In the whole picture compression system, parts other than quantization table 4 are the same as the configuration of drawing 1 .
[0103] The quantization table 4 has the basis child-ized table memory 11, a multiplier 12, a multiplexer 71, and a divider 15. Memory 11 memorizes the basis child-ized table Q. A multiplexer 71 supplies either the 1st scale factor SFa or the 2nd scale factor SFb to a multiplier 12 under control of a controller 8. A multiplier 12 performs the multiplication of the basis child-ized table Q and a scale factor SF, and outputs Q-SFa or Q-SFb.
[0104] If a multiplexer 71 supplies a scale factor SFa to a multiplier 12, a divider 15 will perform the following operations and will output the quantization data Ruv. Rolling round means integer-ization to the nearest integer, and the DCT multiplier Fuv is a multiplier memorized by the DCT coefficient memory 3.
[0105]
Ruv=round [Fuv/(SFa-Quv)]
It encodes in the coding section 5 and the amount NA of code data counts the quantization data Ruv with the counter A of the amount count area 6 of code data after that. The amount NA of code data is the

amount of code data when compressing using the 1st scale factor SFa.
[0106] If a multiplexer 71 supplies a scale factor SFb to a multiplier 12, a divider 15 will perform the following operations and will output the quantization data Ruv.
[0107]
Ruv=round [Fuv/(SFb-Quv)]
It encodes in the coding section 5 and the amount NB of code data counts the quantization data Ruv with the counter B of the amount count area 6 of code data after that. The amount NB of code data is the amount of code data when compressing using the 2nd scale factor SFb.
[0108] Unlike the quantization section of drawing 1 , the quantization section 4 has only the memory 11 for memorizing the basis child-ized table Q, and does not have the memory for memorizing the value which multiplied by the quantization table Q and the scale factor SF. Since the picture compression system of drawing 11 can reduce memory space compared with the picture compression system of drawing 1 , it can aim at miniaturization of a system, and reduction of cost.
[0109] However, after a multiplexer 71 supplies a scale factor SF to a multiplier 12 and calculating with a multiplier 12, Q-SFa or Q-SFb is supplied to a divider 15. That is, only one cushion is in timing by the operation of a multiplier 12. It is necessary to adjust the delay of the timing.
[0110] Since the data which read the picture compression system of drawing 1 from the quantization table memory 13 or the quantization table memory 14 to it are supplied to the direct divider 15, adjustment of timing is easy and the delay of an operation is not produced.
[0111] Next, the example which performs statistics processing only about a sample block is explained. All blocks of the image of one frame were processed in the above-mentioned statistics processing. However, since statistics processing is for estimating the amount of code data to the last, it not necessarily needs to process no blocks. Then, not all blocks can be processed, but only a sample block can be processed, and compaction of the processing time can be aimed at.
[0112] Drawing 10 (A) - (C) shows the example of a sample of a sample block. Drawing shows the assembly of a two-dimensional block. The block which gave the slash is a sample block.
[0113] Drawing 10 (A) shows a checkered sample block. Drawing 10 (B) shows a vertical stripe-like block [sample]. Drawing 10 (C) shows a horizontal stripe-like block [sample]. For example, in the case of drawing 10 (A), statistics processing is alternately performed in order of block 1, block 3, block 5, and ...
[0114] These sample blocks are the number of the one half of a whole block. If statistics processing is performed only about these sample blocks, time amount of statistics processing can be made into abbreviation one half. However, the amount of the code data computed by statistics processing is the one half of the amount of the code data of the image of one frame. In consideration of the amount of code data being one half, it is necessary to determine a scale factor SFx.
[0115] According to this example, two statistics processings can be substantially performed within the time amount of one compression processing about an image of one frame. Statistics processing can search for the optimal scale factor with a sufficient high speed and precision. A picture compression system can generate the code data of the target amount of data with a high speed and a sufficient precision.
[0116] In addition, in statistics processing, although the case where two kinds of quantization tables were used was explained, three or more kinds of quantization tables may be used. However, it is desirable that it is within the limits which quantization processing and coding processing complete between the DCT processing times about 1 block. Usually, if it is the case where two kinds of quantization tables are used, it will be settled in the time amount of DCT processing. If the class of quantization table to be used is increased, it can ask for the highly precise statistics processing SFx, i.e., the more nearly optimal scale factor.
[0117] Moreover, although how to change a scale factor was explained as an approach of adjusting condensation, the approach of it not being based on a scale factor but changing the quantization table itself may be used, and the other approaches may be used.
[0118] Although this invention was explained in accordance with the example above, this invention is

not restricted to these. For example, probably, it will be obvious to this contractor for various modification, amelioration, combination, etc. to be possible.

[0119]

[Effect of the Invention] After a DCT means finishes generating the DCT multiplier of the 1st block, the process at which a code data generation means generates code data with two kinds of condensation, and the process at which a DCT means generates the DCT multiplier of the 2nd block can be made to arrange in parallel according to this invention, as explained above. By making the process concerned parallelize, improvement in the speed of compression processing can be attained. Moreover, highly precise fixed-length-ized processing can be performed by generating code data with at least two kinds of condensation.

[Translation done.]

DESCRIPTION OF DRAWINGS

[Brief Description of the Drawings]
[Drawing 1] It is the block diagram showing the picture compression structure of a system by the example of this invention.
[Drawing 2] It is the flow chart which shows the procedure which the picture compression system by this example performs.
[Drawing 3] It is drawing showing the example of the basis child-ized table of drawing 1 .
[Drawing 4] It is drawing showing how the amount count area of code data of drawing 1 counts the amount of code data a .
[Drawing 5] It is a graph for explaining processing of the scale-factor decision section of drawing 1 .
[Drawing 6] A picture compression system is the timing chart which shows the timing of a processing section unit which performs statistics processing.
[Drawing 7] A picture compression system is the timing chart which shows the timing of a block unit which performs statistics processing.
[Drawing 8] A picture compression system is the timing chart which shows the timing which performs compression processing.
[Drawing 9] It is drawing showing other examples of the amount count area of code data of drawing 1 .
[Drawing 10] The example of a sample of a sample block is shown. Drawing 10 R> 0 (A) is drawing in which a checkered sample block and drawing 10 (B) show a vertical stripe-like block [sample], and drawing 10 (C) shows a horizontal stripe-like block [sample].
[Drawing 11] It is drawing showing other examples of the quantization section of drawing 1 .
[Description of Notations]
1 Image Memory
2 Discrete Cosine Transform (DCT) Section
3 DCT Coefficient Memory
4 Quantization Section
5 Coding Section
6 The Amount Count Area of Code Data
7 Scale-Factor Decision Section
8 Controller
11 Basis Child-ized Table Memory
12 Multiplier
13 Quantization Table A Memory
14 Quantization Table B Memory
15 Divider
61 Counter
71 Multiplexer

[Translation done.]

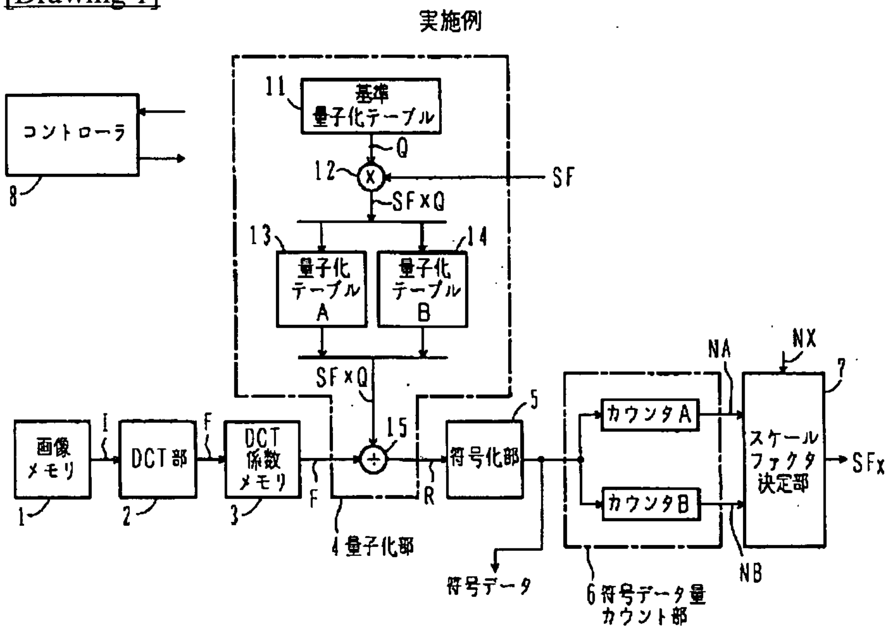
* NOTICES *

JPO and NCIPI are not responsible for any
damages caused by the use of this translation.

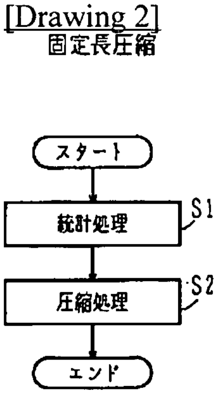
1.This document has been translated by computer. So the translation may not reflect the original precisely.
2.**** shows the word which can not be translated.
3.In the drawings, any words are not translated.

DRAWINGS

[Drawing 1]



[Drawing 2]



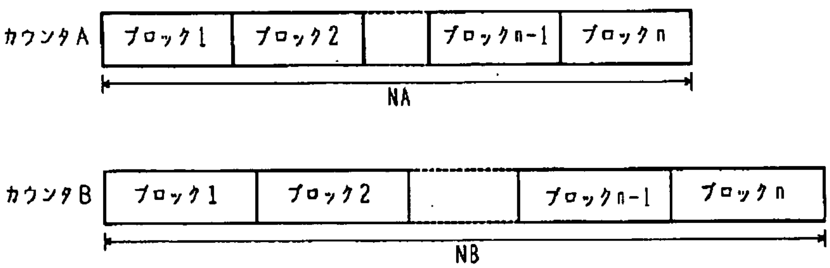
[Drawing 3]

基準量子化テーブル

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

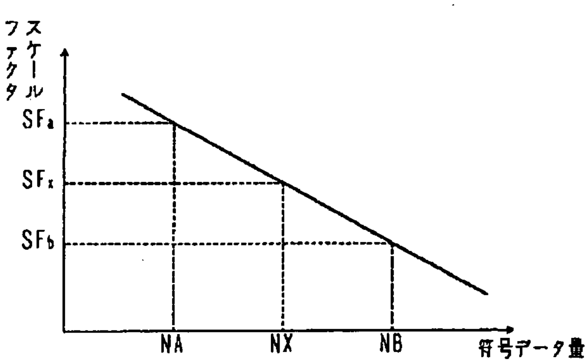
[Drawing 4]

符号データ量カウント部



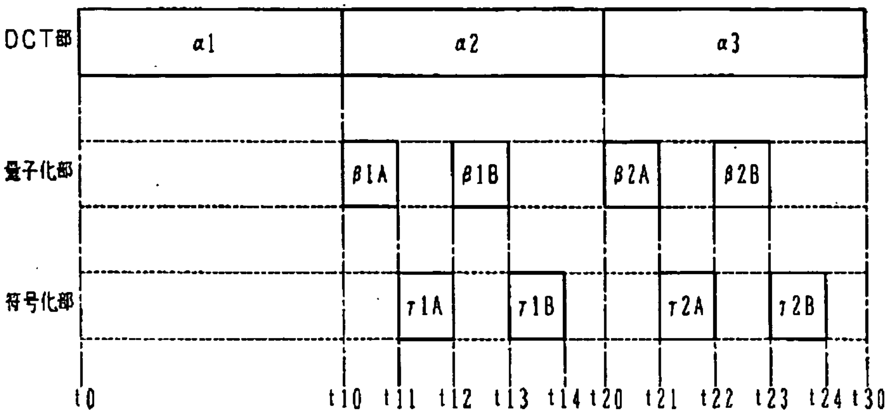
[Drawing 5]

スケールファクタ決定部



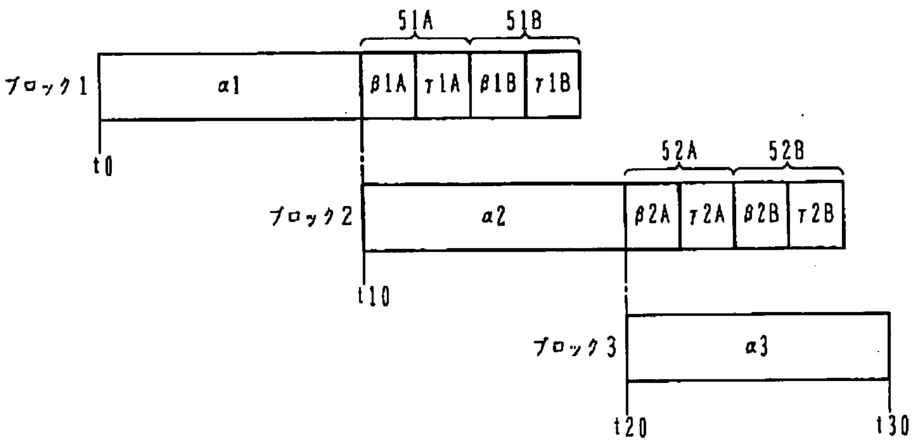
[Drawing 6]

統計処理（処理部毎のタイミング）



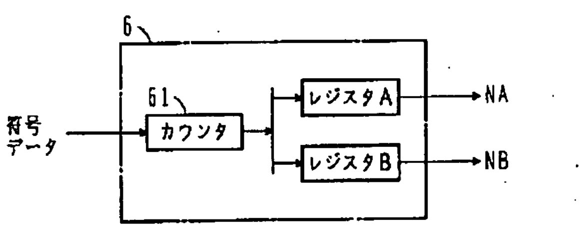
[Drawing 7]

統計処理（ブロック毎のタイミング）



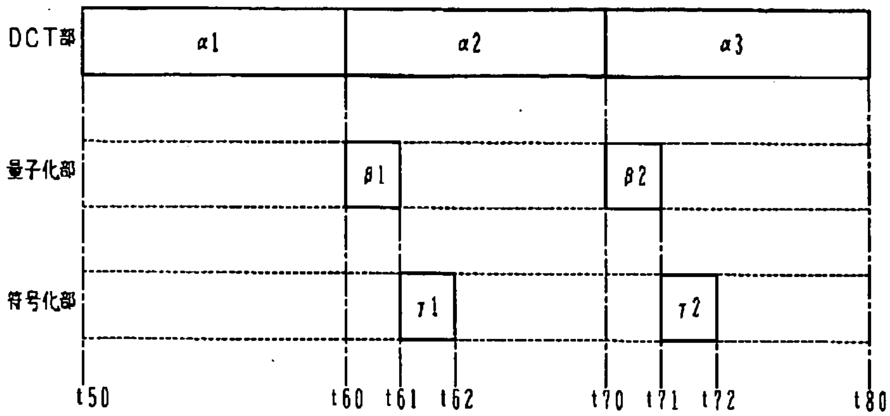
[Drawing 9]

符号データ量カウント部の他の例



[Drawing 8]

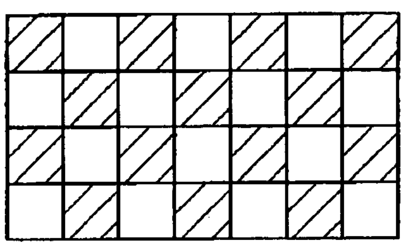
圧縮処理



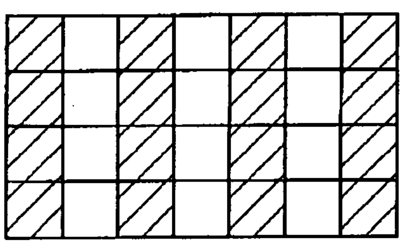
[Drawing 10]

サンプルブロック

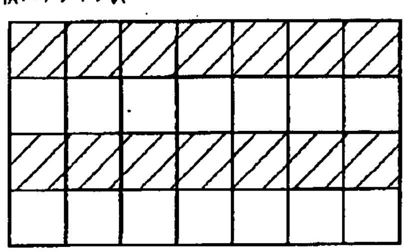
(A) 市松模様状



(B) 縦ストライプ状

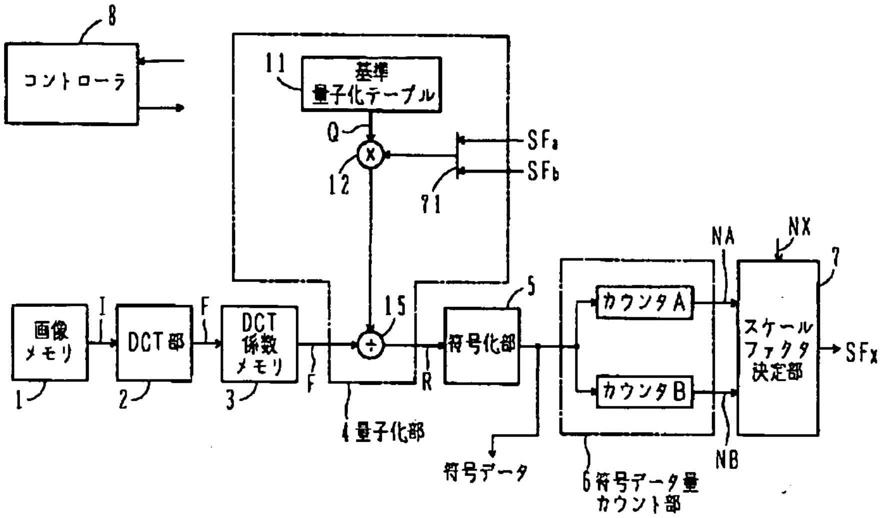


(C) 横ストライプ状



[Drawing 11]

他の実施例



[Translation done.]